# SOFTWARE ENGINEERING

Software engineering is the creation of software using a process similar to other engineering disciplines. It allows for software to be reliable and developed within time and cost estimates. The software engineering curriculum prepares students for a career in reliable, economical software development.

Programming is only one phase (construction) of software engineering. There are many other aspects of the software engineering process, such as requirements definition, architectural design, and quality assurance, which need to be applied in order to develop reliable software on time and within budget constraints. The software engineering curriculum provides students a solid background in both the theory and practice of all phases in the software engineering process, beginning with their first course of study in the Department of Computer Science and Software Engineering, and continuing to the end of the senior year.

Since software is a non-physical product developed and executed on computers, the software engineering curriculum has computer science as its primary engineering science. The computer science courses taken by software engineering majors include the study of algorithms, data structures, database concepts, computer architecture, programming languages and operating systems. Software engineering majors also complete important courses in other closely related fields, such as discrete mathematics, digital logic design, and engineering statistics.

Coverage of software engineering topics begins in a three-term introduction to software development during the freshman and sophomore years. This study continues with coverage of core software engineering areas in the junior year, including software requirements, software architecture, software design, software project management, software construction, software maintenance, software evolution, software quality assurance, and formal methods in software specification and design. All of these courses include individual and team projects relevant to that particular area of software engineering. These projects generally include both written and oral presentations, building upon a technical communication course which introduces the student to the skills necessary for this important aspect of being a software professional. Throughout the senior year, a capstone team project develops and delivers software for a “real-world” client, which is put on display locally at a public exposition.

Courses in various computer science topics such as computer graphics, artificial intelligence, computer networks, computer vision, web-based information systems, and cryptography are among those available as advanced electives. In addition, free elective courses allow students to tailor their undergraduate education to their specific goals.

The student chapter of the Association for Computing Machinery provides seminars and other technical activities throughout the year. The national honor society in the computing and engineering disciplines, Upsilon Pi Epsilon and Tau Beta Pi, both have chapters at Rose-Hulman. Software engineering majors are also eligible to join the Order of the Engineer, which focuses on the ethical and professional responsibilities of an engineer, during the spring of their last year of study.

The Computer Science and Software Engineering faculty strives to maintain an open atmosphere that encourages mutual respect and support as well as learning and sharing of knowledge.

The software engineering program is accredited by the Engineering Accreditation Commission of ABET, https://www.abet.org, under the commission’s General Criteria and Program Criteria for Software and Similarly Named Engineering Programs.

## Requirements

**Summary of graduation requirements for the software engineering major**

To complete the major in software engineering a student must complete the following:

1. All required courses listed by number in the schedule of courses above:

| Code | Title | Hours |
|---|---|---|
| CSSE 120 | Introduction to Software Development | 4 |
| CSSE 132 | Introduction to Systems Programming | 4 |
| CSSE 220 | Object-Oriented Software Development | 4 |
| CSSE 230 | Data Structures and Algorithm Analysis | 4 |
| CSSE 232 | Computer Architecture I | 4 |
| CSSE 280 | Introduction to Web Programming | 4 |
| CSSE 304 | Programming Language Concepts | 4 |
| CSSE 332 | Operating Systems | 4 |
| CSSE 333 | Intro to Database Systems | 4 |
| CSSE 371 | Software Requirements Engineering | 4 |
| CSSE 372 | Software Project Management | 4 |
| CSSE 373 | Formal Methods in Specification and Design | 4 |
| CSSE 374 | Software Design | 4 |
| CSSE 375 | Software Construction and Evolution | 4 |
| CSSE 376 | Software Quality Assurance | 4 |
| CSSE 477 | Software Architecture | 4 |
| CSSE 497 | Senior Capstone Project I | 4 |
| CSSE 498 | Senior Capstone Project II | 4 |
| CSSE 499 | Senior Capstone Project III | 4 |
| ECE 233 | Introduction to Digital Systems | 4 |
| MA 111 | Calculus I | 5 |
| MA 112 | Calculus II | 5 |
| MA 113 | Calculus III | 5 |
| MA 221 | Matrix Algebra & Differential Equations I | 4 |
| MA 276 | Introduction to Proofs | 4 |
| MA 374 | Combinatorics | 4 |
| MA 381 | Introduction to Probability with Applications to Statistics | 4 |
| PH 111 | Physics I | 4 |
| PH 112 | Physics II | 4 |
| CHEM 111 | General Chemistry I | 3 |
| CHEM 111L | General Chemistry I Lab | 1 |
| HUM H190 | First-Year Writing Seminar | 4 |
| ENGL H290 | Technical & Professional Communication | 4 |
| RHIT 100 | Foundations for Rose-Hulman Success | 1 |

2. Eight credits of additional software engineering courses numbered between 300 and 492 and designated as software engineering electives. The student’s academic advisor must approve the course used to satisfy this requirement. Use of software engineering courses numbered 490 through 492 to fulfill this requirement must be approved by the department head. Credits used to satisfy any requirements for a minor or secondary major pursued by a student cannot also be used to satisfy SE elective requirements for the

student's primary or secondary major in Software Engineering. Credits used by a student pursuing a secondary major in SE that are intended to satisfy the SE elective requirement can only be used to satisfy technical or free elective requirements within the student's primary major or not used towards any requirements within the primary major.

3. Four additional credits of technical electives, consisting of any courses in biology, chemistry, engineering (except software engineering and engineering management), geology, mathematics, biomathematics, or physics.
4. Four additional credits of courses offered by the Department of Mathematics excluding MA 351 Problem Solving Seminar MA 356 Problem Solving Seminar. The student's academic advisor must approve the coursesused to satisfy this requirement.
5. Four credits of science electives, which can be any CHEM, GEOL, PH, or BIO courses not already required for the software engineering major.
6. Twenty-eight credits of additional courses offered by the Department of Humanities and Social Sciences; the distribution of these courses must meet the requirements of that department.
7. Twelve credits of free elective courses.These courses must have the approval of the student's academic advisor. Free electives may be selected from any Rose-Hulman course.
8. A total of 192 credits.

# Plan of Study

*Below is a sample plan of study that illustrates one way to achieve the program requirements. Any given student's plan of study may differ based on a variety of factors (e.g., advanced credit, placement exams, adding a minor). Enrolled students will work with their academic advisor; utilize the degree audit/planner to create a specific plan of study.*

| Course | Title | Hours |
|---|---|---|
| **Freshman** | | |
| **Fall** | | |
| CSSE 120 | Introduction to Software Development | 4 |
| MA 111 | Calculus I | 5 |
| PH 111 | Physics I | 4 |
| HUM H190 | First-Year Writing Seminar | 4 |
| RHIT 100 | Foundations for Rose-Hulman Success | 1 |
| | **Hours** | **18** |
| **Winter** | | |
| CSSE 220 | Object-Oriented Software Development | 4 |
| MA 112 | Calculus II | 5 |
| PH 112 | Physics II | 4 |
| HSSA Elective | | 4 |
| | **Hours** | **17** |
| **Spring** | | |
| CSSE 132 | Introduction to Systems Programming | 4 |
| MA 113 | Calculus III | 5 |
| ECE 233 | Introduction to Digital Systems | 4 |
| Science Elective | | 4 |
| | **Hours** | **17** |
| **Sophomore** | | |
| **Fall** | | |
| CSSE 232 | Computer Architecture I | 4 |
| CSSE 280 | Introduction to Web Programming | 4 |
| MA 221 | Matrix Algebra & Differential Equations I | 4 |
| MA 276 | Introduction to Proofs | 4 |
| | **Hours** | **16** |
| **Winter** | | |
| CSSE 230 | Data Structures and Algorithm Analysis | 4 |
| MA 374 | Combinatorics | 4 |
| ENGL H290 | Technical & Professional Communication | 4 |
| CSSE 332 | Operating Systems | 4 |
| | **Hours** | **16** |
| **Spring** | | |
| CSSE 376 | Software Quality Assurance | 4 |
| MA Elective | | 4 |
| CSSE 333 | Intro to Database Systems | 4 |
| HSSA Elective | | 4 |
| | **Hours** | **16** |
| **Junior** | | |
| **Fall** | | |
| CSSE 371 | Software Requirements Engineering | 4 |
| CSSE 372 | Software Project Management | 4 |
| MA 381 | Introduction to Probability with Applications to Statistics | 4 |
| CHEM 111 | General Chemistry I | 3 |
| CHEM 111L | General Chemistry I Lab | 1 |
| | **Hours** | **16** |
| **Winter** | | |
| CSSE 374 | Software Design | 4 |
| CSSE 304 | Programming Language Concepts | 4 |
| Tech Elective | | 4 |
| HSSA Elective | | 4 |
| | **Hours** | **16** |
| **Spring** | | |
| CSSE 373 | Formal Methods in Specification and Design | 4 |
| CSSE 375 | Software Construction and Evolution | 4 |
| HSSA Elective | | 4 |
| Free elective | | 4 |
| | **Hours** | **16** |
| **Senior** | | |
| **Fall** | | |
| CSSE 477 | Software Architecture | 4 |
| CSSE 497 | Senior Capstone Project I | 4 |
| HSSA Elective | | 4 |
| SE Elective | | 4 |
| | **Hours** | **16** |
| **Winter** | | |
| CSSE 498 | Senior Capstone Project II | 4 |
| HSSA Elective | | 4 |
| Free Elective | | 4 |
| SE Elective | | 4 |
| | **Hours** | **16** |
| **Spring** | | |
| CSSE 499 | Senior Capstone Project III | 4 |
| HSSA Elective | | 4 |
| Free Elective | | 4 |
| | **Hours** | **12** |
| | **Total Hours** | **192** |

# Program Objectives
# Software Engineering Program Educational Objectives

The software engineering program prepares its graduates for many types of careers in the computing industry as well as for graduate study in software engineering and in closely related disciplines. Within a few years

after completing the software engineering degree program, our graduates will:

1. Advance beyond their entry-level position to more responsible roles, or progress towards completion of advanced degree(s).
2. Continue to keep pace with advancements in their disciplines, and develop professionally in response to changes in roles and responsibilities.
3. Demonstrate that they can collaborate professionally within or outside of their disciplines at local, regional, national, or international levels.
4. Contribute to the body of computing products, services, or knowledge.
5. Serve as business or technical leaders in organizations, industries, and disciplines.

# Learning Outcomes
# Software Engineering Student Outcomes

By the time students graduate with a Software Engineering degree from Rose-Hulman, they will be able to:

1. an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics
2. an ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors
3. an ability to communicate effectively with a range of audiences
4. an ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts
5. an ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives
6. an ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions
7. an ability to acquire and apply new knowledge as needed, using appropriate learning strategies.